Received: 10 January 2021 | Revised: 15 November 2021 | Accepted: 2 January 2022

DOI: 10.1002/acs.3380

RESEARCH ARTICLE

WILEY

MIMO system identification using common denominator and numerators with known degrees

Antonio Fazzi[1] | **Benjamin Grossmann**[1,2] | **Guillaume Mercère**[2] | **Ivan Markovsky**[1,3,4]

[1]ELEC, Vrije Universiteit Brussel, Brussels, Belgium

[2]Laboratoire d'Informatique et d'Automatique pour les Systèmes, Université de Poitiers, Poitiers, France

[3]International Centre for Numerical Methods in Engineering (CIMNE), Barcelona, Spain

[4]Catalan Institution for Research and Advanced Studies (ICREA), Barcelona, Spain

Correspondence
Antonio Fazzi, ELEC, Vrije Universiteit Brussel, Brussels, Belgium.
Email: Antonio.Fazzi@vub.be

Summary

In system identification, prior knowledge about the model structure may be available. However, imposing this structure on the identified model may be nontrivial. A new discrete-time linear time-invariant identification method is presented in the article that imposes prior knowledge of the degree of the common denominator of the system's transfer function matrix and the degrees of the numerators. First, a method is outlined for the solution in case of exact data. Then, this method is extended for noisy data in the output error setting. An initial estimate obtained by a subspace method is improved by a structured low-rank approximation method. The performance of the method imposing the structure is compared on simulated data with the performance of classical identification methods that do not impose the structure.

KEYWORDS

discrete-time systems, structural prior, structured low-rank approximation, subspace methods, system identification

1 | INTRODUCTION

System identification refers to the problem of determining a mathematical model for a dynamical system from observed data.[1] In this article, we consider discrete-time linear time-invariant (LTI) systems. Even in this case, the resulting system identification problems are often non-convex optimization problem; for instance, maximum likelihood identification problems in the output error and errors-invariables setting are non-convex.[2] This is problematic because iterative non-convex optimization methods are only guaranteed to yield a locally optimal solution. A common approach for addressing this problem is to use regularization or to use convex relaxations in order to provide good initial estimates for further optimization.[3]

Among the convex relaxation methods, subspace methods such as N4SID[4] and MOESP[5] are appealing in that they provide an estimate that can be attained with robust linear algebra tools (e.g., QR factorization and SVD), are easily adaptable to arbitrary numbers of inputs and outputs, and lead to fully-parameterized and well-conditioned black-box state-space forms.[3] However, the "black-box" nature of subspace methods is also a weakness: it is often very difficult to introduce prior information into the system identification procedure.[6] Indeed, this situation is emblematic of the state of the art: there is a need for tools that allow for the incorporation of prior information into the system identification process.[3]

In light of this, there is a growing interest in variations of these subspace methods that allow users to incorporate prior information,[7-11] and the subspace method that is introduced in this article is another such variation. Specifically, the

method described in this article is built around the "common denominator model" (also known as the "scalar matrix fraction model"[12]) and prior information regarding the numerator degrees in the associated transfer function matrix (TFM). This model is often used, for instance, in modal analysis,[12] since MIMO transfer functions with "common denominator" are closely related to minimal state-space realizations. The importance of such minimal realizations is the fact that they are equivalent to reachable and observable systems. Multi-agent systems are also associated with "common denominator" transfer functions/minimal state space realizations.[13]

The approach proposed in this article consists of two steps. The first step consists of the subspace method mentioned above, which leads to a good initial estimate of the parameters that define the system. The second step is a refinement of this initial estimate through the consideration of a structured low-rank approximation (SLRA) problem[14,15] corresponding to system identification. Both of these methods are tailored to the particular nature of the prior information. Thus, the results of system identification are not only more accurate (as is indicated by our experimental results in Section 5), but also interpretable within the "common denominator" framework.

There are widely available tools for system identification that, while they do not take advantage of the particular form of the prior information associated with the common denominator representation, are useful. In this manuscript, the performance of the approach proposed (both with and without the second step) is compared with that of the N4SID identification algorithm.

The article is structured as follows: Section 2 provides a brief background and detailed statement of the system identification problem to be investigated. The matrix-based reformulation of the problem and the SLRA approach are explained in Section 3, and the subspace approach is explained in Section 4. Section 5 presents the simulations performed to assess algorithm performance and discusses the results. Finally, conclusions are offered in the final section.

2 | BACKGROUND AND PROBLEM DESCRIPTION

Let q denote a positive integer, and let $\mathbb{N} = \{0, 1, 2, \dots\}$. For a trajectory $f : \mathbb{N} \to \mathbb{R}^q$, σ denotes the "shift operator" defined by $(\sigma f)(t) = f(t+1)$ for $t \in \mathbb{N}$. A discrete-time, LTI system $\mathcal{S}$ is a set of trajectories $f : \mathbb{N} \to \mathbb{R}^q$ that is linear (i.e., if $f_1, f_2 \in \mathcal{S}$, then so is $a_1 f_1 + a_2 f_2$ for $a_1, a_2 \in \mathbb{R}$) and time-invariant (i.e., if f is in $\mathcal{S}$, then so is the shifted trajectory σf). Often, the trajectories of $\mathcal{S}$ are partitioned into an input $u : \mathbb{N} \to \mathbb{R}^m$ and an output $y : \mathbb{N} \to \mathbb{R}^p$ for some positive integers m, p with $q = m + p$. With this established, $\mathcal{S}$ always[12] has a "polynomial matrix description," which is to say that it can be written as the of the set of trajectories (u, y) for which

$$P(\sigma)\, y = Q(\sigma)\, u, \tag{1}$$

where $P(z), Q(z)$ denote matrix polynomials in the indeterminate z, with

$$Q(z) = Q_0 + Q_1 z + \cdots + Q_{d_Q^{\max}} z^{d_Q^{\max}} \in \mathbb{R}^{p \times m}[z]. \tag{2}$$

In the case that there is no cancellation, it can equivalently be said that the TFM has the form[16] $H(z) = [P(z)]^{-1} Q(z)$. More specifically, the LTI systems to be considered in this article have a "common denominator representation." That is, $\mathcal{S}$ must have a polynomial matrix description in which $P(z) = a(z) I_p$, where

$$a(z) = a_0 + a_1 z + \cdots + a_{d_P - 1} z^{d_P - 1} + z^{d_P} \in \mathbb{R}[z], \tag{3}$$

and I_p is the identity matrix of size p, or equivalently, $H(z) = \frac{1}{a(z)} Q(z)$. That is, $a(z)$ is the common denominator of all entries of the TFM. It will be assumed furthermore that the transfer function is proper, that is, that $d_Q^{\max} \leq d_P$. For each $1 \leq i \leq p$ and $1 \leq j \leq m$, $d_Q(i, j)$ denotes the degree of the i, j entry of $Q(z)$. In particular, this means that the degree $d_Q^{\max}$ of the polynomial $Q(z)$ is given by

$$d_Q^{\max} = \max_{1 \leq i \leq p, 1 \leq j \leq m} d_Q(i, j).$$

This article is concerned in particular with the problem of system identification. That is: given an observed trajectory of an LTI system with a common denominator representation of a known form, the goal is to select the parameters (the coefficients in Equations (2) and (3)) for which the associated TFM comes "as close as possible" to describing the true

system. More specifically: given an integer d_P and a $p \times m$ matrix d_Q, let $\mathcal{M}^{\text{cd}}_{d_P,d_Q}$ denote the set of all LTI systems that have a common denominator representation whose common denominator has degree d_P and whose numerators have degrees $d_Q(i,j)$ for $1 \le i \le p$ and $1 \le j \le m$. Let $\text{err}(\hat{f},f)$ denote a metric for the error in approximation between a time series f and its approximation $\hat{f}$. In this article, this error metric is taken to be

$$\text{err}(f,\hat{f}) = \|\mathcal{H}_{d_P+1}(\hat{f}) - \mathcal{H}_{d_P+1}(f)\|_F, \tag{4}$$

where $\mathcal{H}_L(f)$ is the block-Hankel matrix (see (6) for the definition). This is the "natural" choice in light of the problem reformulation explained in Section 3.1. This error metric can be rewritten as $\text{err}(f,\hat{f}) = \left(\sum_{t=0}^{T-1} w_t(\hat{f}(t) - f(t))^2\right)^{1/2}$, where $w_t = \min\{t+1, d_P+1, T-t+1\}$. For an LTI system $\mathcal{S}$ and an integer $T \ge 1$, let $\mathcal{S}_T$ denote the set of all length-T time series $(f(0), f(1), \ldots, f(T-1))$ associated with a trajectory $f \in \mathcal{S}$.

Problem 1 (Output error). *Given a system for which:*

1. *the system is known to have a common-denominator representation with denominator-degree d_P and numerator-degrees given by the matrix d_Q,*
2. *there is a known trajectory of (exact) inputs $(u_d(0), u_d(1), \ldots, u_d(T-1))$, where $u_d(i) \in \mathbb{R}^m$ for $i = 0, \ldots, T-1$.*
3. *there is a trajectory of noisy measurements of the outputs corresponding to the given inputs given by $(y_d(0), y_d(1), \ldots, y_d(T-1))$, where $y_d(i) \in \mathbb{R}^p$ for $i = 0, \ldots, T-1$.*

The objective is to find the system $\mathcal{S} \in \mathcal{M}^{\text{cd}}_{d_P,d_Q}$ at which the minimum

$$\min_{\mathcal{S},\hat{y}} \text{err}(\hat{y}, y_d) \ \text{ subject to } \ \mathcal{S} \in \mathcal{M}^{\text{cd}}_{d_P,d_Q} \ \text{ and } \ \hat{y} \in \mathcal{S} \tag{5}$$

is attained.

3 | SLRA APPROACH

3.1 | Problem reformulation

An approach to solve Problem 1 via structured low-rank approximation (SLRA) arises naturally from the formulation of the problem. This is because of the condition that the block-Hankel matrix built from the input and output trajectories u_d, y_d is rank-deficient. In particular, for a time series $(f(0), f(1), \ldots, f(T-1))$, let $\mathcal{H}_L(f)$ denote the block-Hankel matrix

$$\mathcal{H}_L(f) = \begin{bmatrix} f(0) & f(1) & \cdots & f(T-L) \\ f(1) & f(2) & \cdots & f(T-L+1) \\ \vdots & \vdots & \ddots & \vdots \\ f(L) & f(L+1) & \cdots & f(T-1) \end{bmatrix}, \tag{6}$$

and for $c \le T-L+1$, let $\mathcal{H}^c_L(f)$ denote the matrix attained by taking the first c columns of $\mathcal{H}_L(f)$. The condition that the discrete-time satisfies common-denominator representation (1) is equivalent to the matrix equation

$$\begin{bmatrix} P_0 & \cdots & P_{d_P-1} & P_{d_P} & Q_0 & Q_1 & \cdots & Q_{d_Q} \end{bmatrix} \begin{bmatrix} \mathcal{H}_{d_P+1}(y_d) \\ -\mathcal{H}^c_{d_Q+1}(u_d) \end{bmatrix} = 0, \tag{7}$$

where $c = T - d_P$ denotes the number of columns in $\mathcal{H}_{d_P+1}(y)$. This can be seen as follows: corresponding to the matrix polynomial $Q(z)$ as in Equation (2), define the associated block matrix Q_{block} to be given by

$$Q_{\text{block}} = \begin{bmatrix} Q_0 & Q_1 & \cdots & Q_{d_Q} \end{bmatrix},$$

and let P_{block} denote the block matrix associated with $P(z)$. Note that $[Q(\sigma)u](t)$ can be rewritten as the matrix product

$$\begin{bmatrix} Q_0 & Q_1 & \cdots & Q_{d_Q} \end{bmatrix} \begin{bmatrix} u(t) \\ u(t+1) \\ \vdots \\ u(t+d_Q) \end{bmatrix} = [Q_{\text{block}} \mathcal{H}_{d_Q+1}(u)](:, t),$$

where, for a matrix M, $M(:, j)$ denotes the jth column of M. Similarly, $[P(\sigma)y](t)$ is the tth column of $P_{\text{block}}\mathcal{H}_{dP+1}(y)$. Thus, the statement that the observed data u, y over the interval $[0, T]$ satisfies the equation $P(\sigma)y = Q(\sigma)u$ over $t = 0, \ldots, T - d_P - 1$ can be rewritten as

$$\begin{aligned} P_{\text{block}}\mathcal{H}_{d_P+1}(y) = Q_{\text{block}}\mathcal{H}_{d_Q+1}(u) \;\Rightarrow \\ P_{\text{block}}\mathcal{H}_{d_P+1}(y) - Q_{\text{block}}\mathcal{H}_{d_Q+1}(u) = 0 \;\Rightarrow \\ \begin{bmatrix} P_{\text{block}} & Q_{\text{block}} \end{bmatrix} \begin{bmatrix} \mathcal{H}_{d_P+1}(y) \\ -\mathcal{H}_{d_Q+1}(u) \end{bmatrix} = 0, \end{aligned}$$

which matches Equation (7). Incorporating the known information to Equation (7) yields

$$\begin{bmatrix} a_0 I_p & \cdots & a_{d_p-1} I_p & I_p & Q_0 & Q_1 & \cdots & Q_{d_Q} \end{bmatrix} \begin{bmatrix} \mathcal{H}_{d_P+1}(y) \\ -\mathcal{H}_{d_Q+1}(u) \end{bmatrix} = 0. \tag{8}$$

Remark 1. The fact that Equation (7) is written in terms of the coefficients from the matrix polynomial representation indicates that this "matrix motivated" approach lends itself to other priors that can be connected to the structure of the matrices P_i and Q_i. For example, one enforces a TFM structure in which each row of the TFM is presented with a common denominator by constraining the matrices P_i to be diagonal.

Let $\text{err}(f, \hat{f})$ be as it is given in Equation (4). Note that because a system in $\mathcal{M}^{\text{cd}}_{d_P, d_Q}$ is completely specified by parameters $\hat{a}, \hat{Q}$, this instance of Problem 1 can be expressed as follows:

Problem 2 (SLRA). *With the information listed in Problem 1, select parameters $\hat{a}(z), \hat{Q}(z)$ with degrees corresponding to d_P, d_Q respectively for which the minimum*

$$\min_{\hat{a}, \hat{Q}, \hat{y}} \|\mathcal{H}_{d_P+1}(\hat{y}) - \mathcal{H}_{d_P+1}(y_d)\|_F \text{ subject to } \begin{bmatrix} P_{\text{block}} & Q_{\text{block}} \end{bmatrix} \begin{bmatrix} \mathcal{H}_{d_P+1}(\hat{y}) \\ -\mathcal{H}^c_{d_Q+1}(u_d) \end{bmatrix} = 0 \tag{9}$$

is achieved.

On the one hand, we can see that this is an SLRA problem: the condition that there exist matrices $P_{\text{block}}, Q_{\text{block}}$ (with P_{block} of full row rank) for which the constraint above holds is equivalent to the statement that $\begin{bmatrix} \mathcal{H}_{d_P+1}(\hat{y}) \\ -\mathcal{H}^c_{d_Q+1}(u_d) \end{bmatrix}$ is of *rank* at most $pd_P + m(d_Q + 1)$. On the other hand, the usage of $\mathcal{H}_{d_P+1}(\hat{y})$ constrains the *structure* of the low-rank matrix being sought, and the relationship between $P_{\text{block}}, Q_{\text{block}}$ and the system parameters constrains the structure of this matrix's left nullspace.

Framing the problem in this way leads to the exciting prospect of applying existing SLRA tools and methods.[14,15] The algorithm described below uses the "variable projection method," but other methods exists such as those based on nuclear norm minimization.[17] Moreover, future approaches could take advantage of software that accounts for missing data.[14,15]

3.2 | Algorithm description

The solution of Problem 2 is sought via an "outer" and "inner" minimization.[14] More specifically: for parameters $\hat{a}, \hat{Q}$, we define

$$R(\hat{a}, \hat{Q}) = \begin{bmatrix} P_{\text{block}} & Q_{\text{block}} \end{bmatrix}, \tag{10}$$

$$f(R) = \min_{y} \|\mathcal{H}(y) - \mathcal{H}(y_d)\|_F \text{ subject to } R(\hat{a}, \hat{Q}) \begin{bmatrix} \mathcal{H}(y) \\ \mathcal{H}(u_d) \end{bmatrix} = 0. \tag{11}$$

The algorithm seeks a minimum by minimizing the "inner minimization" function f. That is, the algorithm attempts to find $\min_{\hat{a},\hat{Q}} f(R(\hat{a}, \hat{Q}))$. The inner minimization function f can be solved directly as a linear least squares problem. The outer minimization can be done by any non-linear optimization method (for instance, Matlab's `fminunc` method). Performance could be improved further using the ideas of Reference 18, that is, computing the gradient directly and efficiently.

However, as was mentioned in the introduction, because Problem 2 is a non-convex optimization, it requires a good initial guess for the parameters. The subspace algorithm described in the next section provides an initial guess in a way that takes advantage of the particular nature of the prior information.

4 | SUBSPACE APPROACH

The goal of the algorithm detailed here is to take the data described in Problem 1 as its inputs and produce the parameters of an estimated system as its output. As with other subspace approaches,[19] we begin by obtaining an estimate for the column-space of the "extended observability matrix" $\mathcal{O}_{d_P+1}$. However, unlike other subspace methods, it does not use this estimate to generate a state-space model for the system, but it directly computes the desired parameters.

4.1 | Algorithm description

For simplicity of discussion, we first reformulate Equation (8) as

$$\begin{bmatrix} a_0 I_p & \cdots & a_{d_P-1} I_p & I_p & Q_0 & Q_1 & \cdots & Q_{d_P} \end{bmatrix} \begin{bmatrix} \mathcal{H}_{d_P+1}(y) \\ -\mathcal{H}_{d_P+1}(u) \end{bmatrix} = 0, \tag{12}$$

where we take $Q_k = 0$ for $k > d_P$.

It is given that $\mathcal{S} \in \mathcal{M}^{\text{cd}}_{d_P,d_Q}$. Let U_p denote a matrix whose columns form an orthonormal basis for the nullspace of $\mathcal{H}_{d_P+1}(u)$. It is known[19] that, under assumption that outputs are subject only to zero-mean, uncorrelated, asymptotically unbiased output error, the matrix $\mathcal{H}_{d_P+1}(y_d)U_p$ gives an "unbiased estimate" of the column-space of $\mathcal{O}_{d_P+1}$, which is[17] equivalently the span of all $y|_{[0,d_P]}$ such that $(0, y) \in \mathcal{S}$. These y satisfy $P(\sigma)y = 0$. That is, each component of y satisfies $a(\sigma)y_i = 0$.

The equation associated with the condition that $a(\sigma)y_i = 0$ can be written as follows: let $\otimes$ denote the standard Kronecker product[20] and π denote the permutation matrix for which $\pi(x \otimes y) = y \otimes x$ for $x \in \mathbb{R}^{d_P+1}$ and $y \in \mathbb{R}^p$ (π is sometimes referred to as the commutation matrix[21] $K^{(p,d_P+1)}$). Let a denote the vector whose entries are the coefficients of $a(z)$, that is, $a = (a_0, \ldots, a_{d_P-1}, 1)$. We find that

$$\begin{aligned} (a^T \otimes I_p)\mathcal{H}(y)U_p = (a^T \otimes I_p)\pi^T\pi\mathcal{H}(y)U_p = 0 \;\Rightarrow \\ [\pi(a \otimes I_p)]^T[\pi\mathcal{H}(y)]U_p = 0 \;\Rightarrow \\ (I_p \otimes a)^T \begin{bmatrix} \mathcal{H}(y_1) \\ \vdots \\ \mathcal{H}(y_p) \end{bmatrix} U_p = 0 \;\Rightarrow \\ \begin{bmatrix} a^T\mathcal{H}(y_1)U_p \\ \vdots \\ a^T\mathcal{H}(y_n)U_p \end{bmatrix} = 0. \end{aligned}$$

That is, the equation to be solved is equivalent to the statement that $a^T\mathcal{H}(y_i)U_p = 0$ for $i = 1, \dots, p$. This system of equations can be written as

$$a^T\begin{bmatrix}\mathcal{H}(y_1)U_p & \cdots & \mathcal{H}(y_p)U_p\end{bmatrix} = 0 \;\Rightarrow\; a^T\mathcal{Y}_p = 0.$$

The coefficients in $a(z)$ are computed using the least squares solution to this system. From there, the Q coefficients for which

$$[P_{\text{block}}(\hat{a}) \;\; Q_{\text{block}}]\begin{bmatrix}\mathcal{H}_{d_P+1}(y)\\ \mathcal{H}_{d_Q+1}(u)\end{bmatrix} = 0$$

are uniquely determined (assuming that the input is persistently exciting[17]), and each row of Q_{block} can be solved separately by solving the associated system of equations.

4.2 | Connection with other subspace methods

In the Multivariable Output-Error State-sPace (MOESP) algorithm,[19] the input is "projected out" via the RQ factorization

$$\begin{bmatrix}\mathcal{H}_{d_P+1}(u_d)\\ \mathcal{H}_{d_P+1}(y_d)\end{bmatrix} = \begin{bmatrix}R_{11} & 0\\ R_{21} & R_{22}\end{bmatrix}\begin{bmatrix}Q_1\\ Q_2\end{bmatrix}. \tag{13}$$

In particular, the algorithm exploits the fact that the column space of R_{22} is equal to that of the height-$(d_P + 1)$ observability matrix $\mathcal{O}_{d_P+1}$.[19]

Our method begins along similar lines: the matrix U_p is equal to Q_2^T. Expanding the matrix product on the right of Equation (13) and noting that $Q_1Q_2^T = 0, Q_2Q_2^T = I$ reveals that

$$\mathcal{H}_{d_P+1}(y_d)Q_2^T = R_{21}Q_1Q_2^T + R_{22}Q_2Q_2^T = R_{22},$$

which is to say that R_{22} is equal to the matrix $\mathcal{H}(y_d)U_p$ considered in the algorithm explanation above. However, rather than using the matrix with the input projected out to relate estimated state data to output data, the subspace algorithm we propose uses this property of the nullspace to compute the desired denominator parameters directly.

As is the case for MOESP, the subspace method presented here can be modified to apply to situations beyond the output-error problem via the introduction of *instrumental variables*. For example, PI-MOESP ("past input" MOESP)[4] accounts for the presence of colored noise by using the RQ factorization

$$\begin{bmatrix}\mathcal{H}_{d_P+1}(u_d^+)\\ Z_N\\ \mathcal{H}_{d_P+1}(y_d^+)\end{bmatrix}\begin{bmatrix}R_{11} & 0 & 0 & 0\\ R_{21} & R_{22} & 0 & 0\\ R_{31} & R_{32} & R_{33} & 0\end{bmatrix}\begin{bmatrix}Q_1\\ Q_2\\ Q_3\\ Q_4\end{bmatrix}, \tag{14}$$

where u^+, y^+ are trajectories measured over $t = d_P, d_P + 1, \dots, T + d_P$, and Z_N denotes the "instrumental variable matrix" $Z_N = \mathcal{H}_{d_P+1}(u^-)$, where u^- denotes a shifted input trajectory measured over $t = 0, 1, \dots, T$. From there, the desired column space of the extended observability matrix, is estimated as that of R_{32}.

The instrumental variable from the PO-MOESP ("past output" MOESP)[4] algorithm can also be incorporated into the subspace method presented in this manuscript.

5 | SIMULATION EXPERIMENT

In the article so far, two viable approaches to system identification have been presented: one is the "subspace method," and the other is the SLRA based method, in which we use the subspace method to generate our initial estimate. In this

section, the performance of the methods proposed above is compared to the performance of N4SID, a well established[4] subspace-based approach to system identification.

5.1 | First simulation experiment

5.1.1 | Data generation protocol

The algorithms are repeatedly applied to a randomly generated trajectory of a 2-input 2-output system with noisy outputs. The system is randomly generated in such a way that the system is necessarily stable and has a denominator representation with d_P, d_Q given by

$$d_P = 5, \quad d_Q = \begin{bmatrix} 1 & 3 \\ 4 & 4 \end{bmatrix}.$$

The identification data for such a system is generated with the following steps:

1. A list of poles of a randomly generated system (using Matlab's `drss`) are produced.
2. The poles are normalized so that each pole z_* satisfies $|z_*| < 0.8$, and the common denominator is set to be the monic polynomial $a(z)$ with these roots.
3. Numerator polynomial coefficients are generated. Each coefficient is independently and identically distributed (i.i.d.) and uniformly random over $[0, 10]$.
4. An i.i.d. uniform random input time series $(u_d(0), u_d(1), \ldots, u_d(T-1))$ is generated with $T = 100$, and the corresponding output $y(0), y(1), \ldots, y(T-1)$ is computed.
5. A zero mean i.i.d. normally distributed noise time series $(\eta(0), \eta(1), \ldots, \eta(T-1))$ is generated and normalized so that the signal to noise ratio (with "signal" y and "noise" η) is 20 dB.
6. The output data for identification is given by $y_d = y + \eta$.

Note that only stable systems are considered because standard subspace methods are known to have trouble with unstable systems.[22,23] The success of each estimate $\hat{H}(z)$ for $H(z) = \frac{1}{a(z)}Q(z)$ is assessed on its "relative H_∞ error." That is, for each estimate $\hat{H}(z)$, the quantity $\|H - \hat{H}\|_\infty / \|H\|_\infty$ is computed, where

$$\|H - \hat{H}\|_\infty = \sup_{\omega \in \mathbb{R}} \sigma_{\max}[H(j\omega) - \hat{H}(j\omega)], \tag{15}$$

with $\sigma_{\max}(M)$ denoting the maximal singular value of the matrix M. A smaller relative error implies a better fit. We also take note of whether or not the estimated transfer function correctly determines that the system under consideration is stable. The closeness of fit for the subspace method and SLRA method are further compared with the following criteria:

1. Pole fit: we compute $\sum_{i,j=1}^{2} \delta_p(h_{ij}, h_{ij})$, where we define δ_p as follows: if $\mathcal{P}_i$ denotes the set of poles of h_i for $i = 1, 2$, then

$$\delta_p(h_1, h_2) = \sum_{p_1 \in \mathcal{P}_1} \min_{p_2 \in \mathcal{P}_2} |p_1 - p_2|^2. \tag{16}$$

2. Parameter error: we compute $\sqrt{\|\hat{a} - a\|^2 + \sum_i \|\hat{Q}_i - Q_i\|_F^2}$.

First, the algorithms were applied to each of 1000 randomly generated sets of system identification data, each with signal to noise ratios (for output measurement) of 20 dB. Then, in order to assess the relationship between performance and output measurement accuracy, the above procedure is applied to 100 sets of system identification data with each of signal to noise ratios 10 dB, 15 dB, $\ldots$, 70 dB.

Remark 2. Unlike the subspace and SLRA methods presented in this article, N4SID does not make use of the prior information regarding the common denominator representation of the system, and it is unlikely to produce a system estimate

TABLE 1 Statistics for identification success over 1000 trials

	H_∞ fit: mean, median	Best H_∞ fit (out of 1000)	Stable model (out of 1000)
Subspace	1.113, 0.195	71	800
Sub + SLRA	0.303, 0.098	804	784
N4SID	1.065, 0.234	125	718

Note: The first column reports the mean and median of the relative H_∞ error among all models. The second column reports the number of trials for which the algorithm gave the smallest relative H_∞ error. The third column reports the number of trials for which the estimated model was stable.

TABLE 2 Subspace and SLRA method comparative success for identification success over 1000 trials

	Pole error: mean, median	Best pole fit	Parameter error: mean, median	Best parameter fit
Subspace	0.819, 0.668	342	43.791, 7.435	172
Sub + SLRA	119.302, 0.534	658	351.853, 6.607	828

that matches this given structure. In particular, it is unlikely that the common-denominator of the resulting transfer function has the correct degree. Thus, there is no obvious way to obtain a parameter estimate (e.g., an estimate for the common denominator) from the system-estimate. For this reason, pole fit and parameter error are not considered for the N4SID estimate.

Remark 3. The N4SID algorithm requires a priori knowledge of the order of the system to be identified. As can be seen as a consequence of Gilbert's diagonal realization,[16] a system with input-dimension m, output-dimension p, common denominator degree d_P, and known numerator degrees will generically have order $\min\{m, p\} \cdot d_P$. As such, a system generated in the manner described above will, with practically 100% probability, have order 10. For this reason, the N4SID estimates obtained for this experiment use the assumption that the order of the desired system is 10.

5.1.2 | Results and discussion

The overall results of part 1 of the experiment are summarized in Table 1. In terms of both mean and median performance, we can see that the SLRA based method did the best. About the median, the subspace method precedes N4SID, while the situation is reversed for the mean value. In the majority of trials, the SLRA-based method yields the closest fit. This all indicates that both the subspace method and the SLRA method outperform N4SID.

The results regarding parameter and pole fit for the subspace and SLRA methods are summarized in Table 2. Again, if one considers median performance and which method produces the best model, the results indicate that the SLRA method is the best. It is notable, however, that the improvement in parameter fit and pole fit due to the addition of the SLRA step is far less dramatic than the improvement to H_∞ fit. However we remark that this is an averaged result, so few samples where the SLRA method had convergence problems could generate misleading results.

It is notable that, in spite of strength of the subspace and SLRA algorithms as proved by median performance, the N4SID method has lower H_∞ error than the subspace method. This seems to indicate that there are rare pathological cases for which the subspace algorithm yields estimates with an H_∞ error that is significantly larger than the median, which results in a dramatic change to the mean but not to the median. The fact that this effect is less dramatic for N4SID may be a fruitful avenue for future research. A similar situation is observed in Table 2 in the comparison between SLRA and the subspace method for the poles and the parameters estimation errors.

As is seen in the last column of Table 1, the algorithms incorrectly yield an unstable system-estimate in about 20% of cases. Interestingly, both the subspace and the SLRA methods did better than N4SID in this regard since the latter *failed* around 30% of the times.

By looking at the histograms in Figures 1 and 2, we can have a better idea of what actually happens. In Figure 1 we see that the SLRA method computes a solution with small relative error in most of the trials, followed by the subspace method. On the other hand, for increasing relative errors, N4SID gains this leadership while SLRA is the last in terms of number of trials. A similar situation can be observed in Figure 2, where SLRA performs better in terms of number of computed solutions with small relative errors.

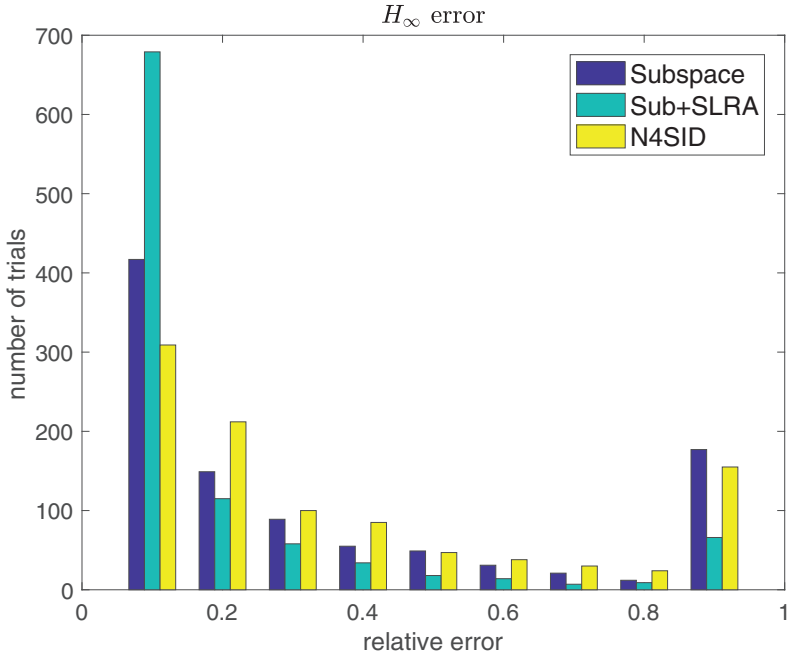

FIGURE 1 Distribution of relative H_∞ error for 1000 experiments

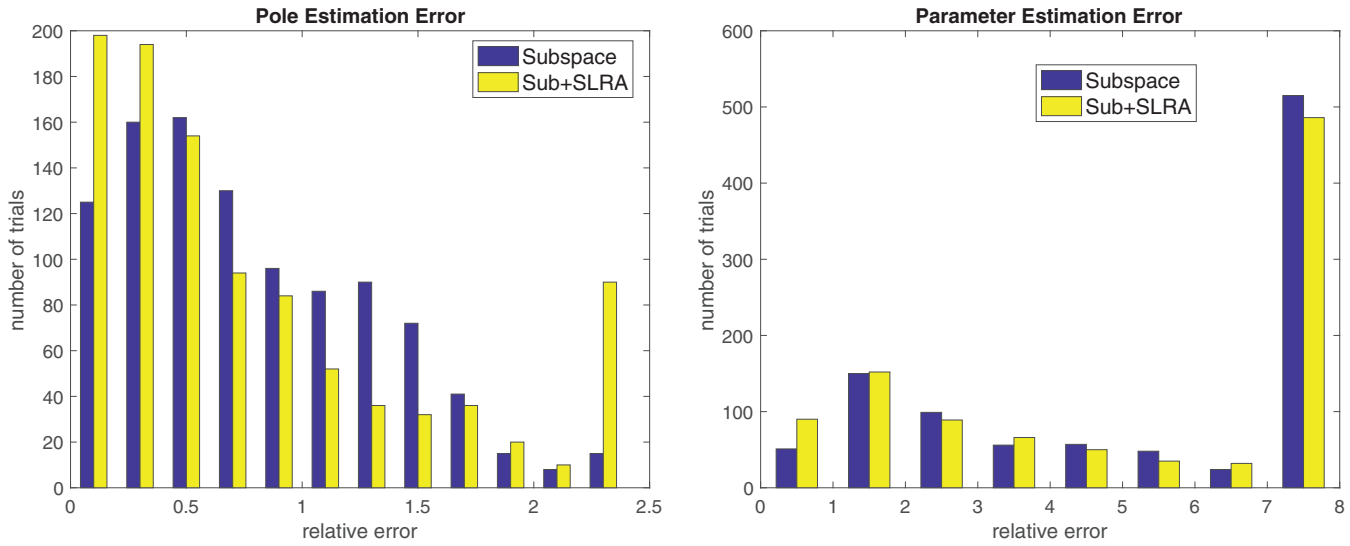

FIGURE 2 Distribution of pole and parameter estimation errors for 1000 experiments

The results of part 2 of the experiment are summarized in Figures 3 and 4. As one would expect, the accuracy of the estimate increases as the amount of measurement error decreases (i.e., the signal to noise ratio increases). Regardless of noise level, the subspace and SLRA methods yield a better fit than N4SID, and the addition of the SLRA step yields an improvement to the subspace estimate. Moreover, the quality of the pole and parameter estimates from the subspace and SLRA methods also improves with increase in the SNR of the output.

5.2 | Second simulation example

We show now an example with 3 inputs and 3 outputs. Following Reference 13, we can imagine that we have three agents in a networked system whose interactions are modeled by the corresponding entries of the transfer function (see Figure 5).

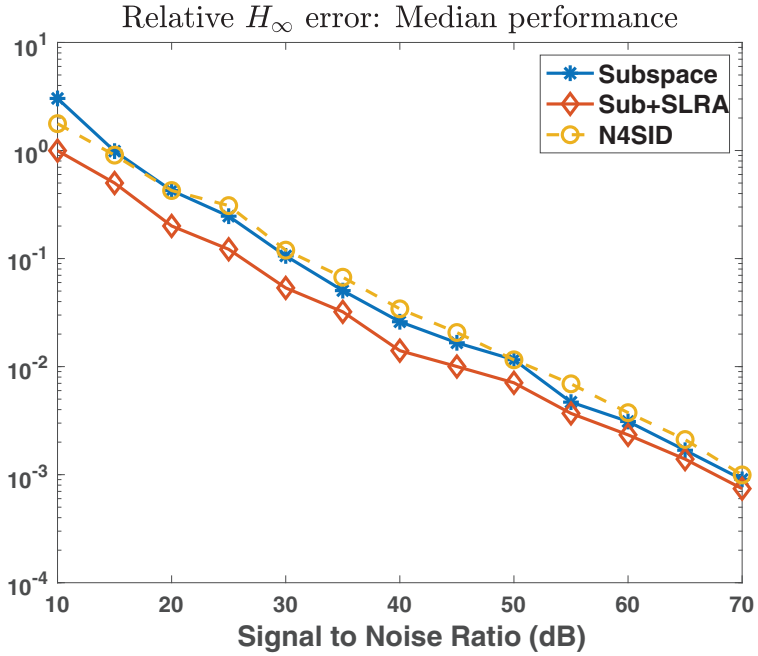

FIGURE 3 Plot of relative H_∞ error against output SNR: Median averaged over 100 experiments

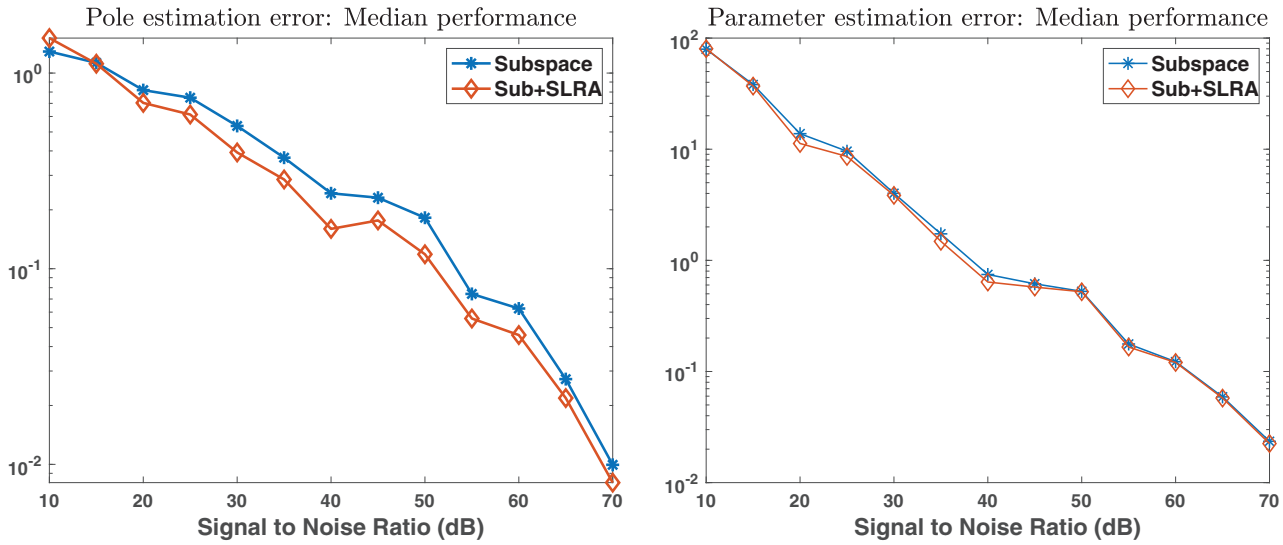

FIGURE 4 Plot of relative pole error and parameter error against output SNR: Median averaged over 100 experiments

The setup for the experiment is the same as the one described in Section 5.1.1, but we restrict to the first part of the experiment only, where we fix the signal to noise ratio to 20 dB and we report the results of the statistics averaged over 1000 randomly generated systems. The degrees of the transfer function polynomials are

$$d_P = 5, \quad d_Q = \begin{bmatrix} 1 & 2 & 1 \\ 3 & 1 & 2 \\ 1 & 1 & 4 \end{bmatrix}.$$

We only need to slightly change point 4 since the data length $T = 100$ turns out to be too small for the N4SID algorithm, which does not work in this case and it returns error messages. The length of the generated data trajectory is increased up to $T = 200$.

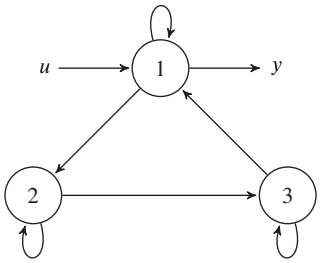

FIGURE 5 Networked system with three agents from Reference 13

TABLE 3 Statistics for identification success over 1000 trials for an example with 3 inputs and 3 outputs

	H_∞ fit: mean, median	Best H_∞ fit (out of 1000)	Stable model (out of 1000)
Subspace	1.370, 0.155	22	858
Sub + SLRA	0.415, 0.084	736	841
N4SID	0.218, 0.187	242	992

Note: The first column reports the mean and median of the relative H_∞ error among all models. The second column reports the number of trials for which the algorithm gave the smallest relative H_∞ error. The third column reports the number of trials for which the estimated model was stable.

TABLE 4 Subspace and SLRA method comparative success for identification success over 1000 trials

	Pole error: mean, median	Best pole fit	Parameter error: mean, median	Best parameter fit
Subspace	2.852, 0.543	329	612.528, 5.324	99
Sub + SLRA	29.113, 0.366	671	619.293, 4.578	901

We show in Tables 3 and 4 the numerical results of the simulation for all the considered errors: H_∞ fit, poles and parameters error computed by the three considered algorithms.

The results about the subspace and the SLRA methods are similar to the previous experiment since SLRA uses the subspace estimate as initial guess; but the numerical performances of N4SID are slightly different. The mean (but not the median) in the computation of the H_∞ fit improves and the number of computed stable models is bigger than 99%. This is probably due to the increase of the length of the sampled trajectory, that is the increase of available data for the identification problem; however the overall performance of the numerical simulation is analogous to the one in Section 5.1.2 (we do not show the corresponding histograms since they are very similar).

An important difference with respect to Section 5.1.2 is the reduction in the gaps between pole and parameter errors (compare Tables 2 and 4). But we need to remark that the current implementation of the SLRA method with structured kernel is inefficient (see Reference 18 for possible improvements which are out of the scope of this article, though), therefore both the bigger dimension of the example as well as the longer simulated trajectory with respect to the previous example contribute to the higher computational time of this solution method.

To sum up the simulation results, plugging in the solution of the subspace method into the SLRA approach actually leads to an improvement in the computation. However, the proposed subspace method itself is worth for several reasons:

1. it is computationally cheap;
2. it is able to exploit the prior information on the degrees of the polynomials in the transfer function;
3. it performs better than N4SID both in terms of median and in the computation of solutions with *small* relative error.

6 | CONCLUSIONS

In this article, two system-identification methods are presented. These methods are designed to produce estimates that are both accurate and interpretable within the "common denominator" framework. The first method is a subspace method,

and the second method tries to improve this initial estimate based on an optimization approach through structured low rank approximation. The accuracy of the estimate produced by these methods is compared to that of N4SID by showing two numerical experiments with synthetic data, and the methods presented indeed outperform N4SID.

ACKNOWLEDGMENT

The research leading to these results has received funding from: the Catalan Institution for Research and Advanced Studies (ICREA), the Fond for Scientific Research Vlaanderen (FWO) projects G090117N and G033822N; and the Fonds de la Recherche Scientifique FNRS–FWO EOS Project 30468160.

ORCID

Antonio Fazzi https://orcid.org/0000-0003-0417-4481
Benjamin Grossmann https://orcid.org/0000-0001-9372-6238
Guillaume Mercère https://orcid.org/0000-0001-9224-4894
Ivan Markovsky https://orcid.org/0000-0001-9976-9685

REFERENCES

1. Ljung L. *System Identification Theory for the User*. 2nd ed. Prentice Hall; 1999.
2. Åström KJ. Maximum likelihood and prediction error methods. *Automatica*. 1980;16:551-574.
3. Ljung L. Perspectives on system identification. *IFAC Proc Vol*. 2008;41:7172-7184.
4. Van Overschee P, De Moor B. *Subspace Identification of Linear Systems: Theory, Implementation, Applications*. Kluwer Academic Publishers; 1996.
5. Verhaegen M. Identification of the deterministic part of MIMO state space models given in innovations form from input output data. *Automatica*. 1994;30:61-74.
6. Mercère G. Regression techniques for subspace-based black-box state-space system identification: an overview; 2013. arXiv preprint. https://arxiv.org/abs/1305.7121v1
7. Trnka P, Havlena V. Subspace like identification incorporating prior information. *Automatica*. 2009;45:1086-1091.
8. Alenany A., Shang H., Soliman M., Ziedan I.. Subspace identification with prior steady-state information. Proceedings of the International Conference on Computer Engineering and Systems; December 2010; Cairo, Egypt.
9. Wang Y, Zhang L, Zhao Y. Improved closed-loop subspace identification with prior information. *Int J Syst Sci*. 2018;49:1821-1835.
10. Hou J, Chen F, Li P, Zhu Z. Prior-knowledge-based subspace identification for batch processes. *J Process Control*. 2019;82:22-30.
11. Markovsky I, Mercère G. Subspace identification with constraints on the impulse response. *Int J Control*. 2017;90(2017):1728-1735.
12. Cauberghe B. *Applied Frequency-Domain System Identification in the Field of Experimental and Operational Modal Analysis*. PhD thesis. Vrije Universiteit Brussel. Brussels, Belgium; 2004.
13. Yu C, Chen J, Li S, Verhaegen M. Identification of affinely parameterized state space models with unknown inputs. *Automatica*. 2020;122:109271. doi:10.1016/j.automatica.2020.109271
14. Markovsky I, Usevich K. Software for weighted structured low-rank approximation. *J Comput Appl Math*. 2014;256:278-292.
15. Usevich K, Markovsky I. Software package for Mosaic-Hankel structured low-rank approximation. Proceedings of the IEEE Conference on Decision and Control; 2019:7165-7170; Nice, France.
16. Kaliath T. *Linear Systems*. Prentice Hall; 1980.
17. Markovsky I. *Low-Rank Approximation: Algorithms, Implementation, Applications*. 2nd ed. Springer International Publishing AG; 2019.
18. Usevich K, Markovsky I. Variable projection for affinely structured low-rank approximation in weighted 2-norms. *J Comput Appl Math*. 2014;272:430-448.
19. Verhaegen M, Verdult V. *Filtering and System Identification, A Least Squares Approach*. Cambridge University Press; 2007.
20. Weisstein EW. Kronecker product. Accessed August 11, 2021. https://mathworld.wolfram.com/KroneckerProduct.html
21. Xu C. Commutation matrices and commutation tensors. *Linear Multilinear Algebra*. 2020;68:1721-1742.
22. Maciejowski JM. Guaranteed stability with subspace methods. *Syst Control Lett*. 1995;26:153-156.
23. Chui NLC, Maciejowski JM. Realization of stable models with subspace methods. *Automatica*. 1996;32:1587-1595.

How to cite this article: Fazzi A, Grossmann B, Mercère G, Markovsky I. MIMO system identification using common denominator and numerators with known degrees. *Int J Adapt Control Signal Process*. 2022;1-12. doi: 10.1002/acs.3380